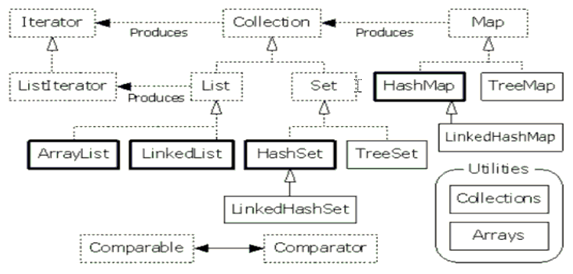
****集合****

集合特点：  
    1. 用于存储对象的容器。  
    2. 集合的长度是可变的。  
    3. 集合中不可以存储基本数据类型值。  
    集合容器因为内部的数据结构不同，有多种具体容器。  
    不断的向上抽取，就形成了集合框架



框架的顶层Collection接口：  
  
    Collection的常见方法：  
    1、添加：  
    boolean add(Object obj);  
    boolean addAll(Collection coll);  
  
    2、删除：  
    boolean remove(Object obj);  
    boolean removeAll(Collection coll);  
    void clear();  
  
    3、判断：  
    boolean contains(Object obj);  
    boolean containsAll(Collection coll);  
    boolean isEmpty();判断集合中是否有元素。  
  
    4、获取：  
    int size();  
    Iterator iterator();  
    取出元素的方式：迭代器。  
    该对象必须依赖于具体容器，因为每一个容器的数据结构都不同，所以该迭代器对象是在容器中进行内部实现的，也就是iterator方法在每个容器中的实现方式是不同的。  
    对于使用容器者而言，具体的实现不重要，只要通过容器获取到该实现的迭代器的对象即可，也就是iterator方法。  
    Iterator接口就是对所有的Collection容器进行元素取出的公共接口。  
  
    5、其他：  
    boolean retainAll(Collection coll);取交集  
    Object toArray();将集合转成数组

****List、Set****  
  
    Collection  
          |--List：有序（存入和取出的顺序一致），元素都有索引（角标），允许重复元素。  
          |--Set：元素不能重复，无序。  
  
    List：特有的常见方法。  
    有一个共性特点就是都可以操作角标。  
    1、添加  
          void add(index,element);  
          void addAll(index,collection);  
  
    2、删除  
          Object remove(index);  
  
    3、修改  
          Object set(index,element);  
  
    4、获取：  
          Object get(index);  
          int indexOf(object);  
          int lastIndexOf(object);  
          List subList(from,to);  
  
    List集合可以完成对元素的增删改查。

List：

1.可以允许重复的对象。

　　  2.可以插入多个null元素。

        3.是一个有序容器，保持了每个元素的插入顺序，输出的顺序就是插入的顺序。

        4.常用的实现类有 ArrayList、LinkedList 和 Vector。ArrayList 最为流行，它提供了使用索引的随意访问，而 LinkedList 则对于经常需要从 List 中添加或删除元素的场合更为合适。

Set：

1.不允许重复对象

　　   2. 无序容器，你无法保证每个元素的存储顺序，TreeSet通过 Comparator  或者 Comparable 维护了一个排序顺序。

        3. 只允许一个 null 元素

        4.Set 接口最流行的几个实现类是 HashSet、LinkedHashSet 以及 TreeSet。最流行的是基于 HashMap 实现的 HashSet；TreeSet 还实现了 SortedSet 接口，因此 TreeSet 是一个根据其 compare() 和 compareTo() 的定义进行排序的有序容器。

map：

1.Map不是collection的子接口或者实现类。Map是一个接口。

2.Map 的 每个 Entry 都持有两个对象，也就是一个键一个值，Map 可能会持有相同的值对象但键对象必须是唯一的。

3. TreeMap 也通过 Comparator  或者 Comparable 维护了一个排序顺序。

4. Map 里你可以拥有随意个 null 值但最多只能有一个 null 键。

5.Map 接口最流行的几个实现类是 HashMap、LinkedHashMap、Hashtable 和 TreeMap。

HashMap和HashTable的区别一种比较简单的回答是：

（1）HashMap是非线程安全的，ConcurrentHashMap是HashMap的优化，HashTable是线程安全的。

（2）HashMap的键和值都允许有null存在，而HashTable和ConcurrentHashMap则都不行。

（3）因为线程安全、哈希效率的问题，HashMap效率比HashTable的要高。